REMARKS

In the aforementioned Office Action, claims 58-73 were examined and rejected. In view of the foregoing amendments and the following remarks, Applicants hereby respectfully request reconsideration of the Application.

An Appendix with a marked-up version of the above amendments follows at the end of this response.

Rejection under 35 U.S.C. § 102

In paragraph 3 of the Office Action, the Examiner rejected claims 58-66 and 71-73 under 35 U.S.C. § 102(e) as being anticipated by Williams (non-patent literature). Applicants respectfully traverse.

Regarding claim 58, the Examiner stated that Williams "discloses a method for mapping a texture onto a surface of a computer generated object comprising the steps of approximating a true pixel color by performing a number of texture operations, texture operations being determined by a **geometric shape of a projection of a pixel on the texture** (pages 1-3)," (emphasis added). As an example of these texture operations, the Examiner stated the Williams "teaches the projection of a flat surface image onto a curved surface ... [t] hus, a true pixel color is generated with his **parametric interpolation**," (emphasis added).

Applicants respectfully submit that projecting a flat surface image onto a curved surface is not the same as using a geometric shape of a projection of a pixel on the texture to determine texturing operations. Williams discloses that sampling difficulties may arise when projecting a flat source image onto a curved surface. Accordingly, Williams proposes a pyramidal parametric

image data structure to minimize those sampling difficulties, and reduce computational requirements, minimize aliasing effects, and assure continuity between target images (Abstract, first two paragraphs). However, Williams does **not** suggest, teach, or disclose "texturing operations being determined by a geometric shape of a projection of a pixel on the texture," as claimed. More specifically, Williams does not disclose projecting a pixel on a texture, or determining a texture operation by a geometric shape of a projection of a pixel on a texture. In fact, projecting a flat source image (i.e., a texture) onto a curved surface as disclosed by Williams is in stark contrast to Applicants' invention of projecting a surface pixel onto a texture.

Williams defines "parametric interpolation" as intra- and inter-level interpolation of a pyramidal data structure, (page 2, section 2, first paragraph). That is, a pyramidal data structure with intra- and inter-level interpolation is referred to as a pyramidal parametric data structure. Williams uses parametric interpolation to produce functional continuity within and between levels of a pyramidal data structure, as specified by a continuous function of parameters (U,V,D), (page 2, section 2, first paragraph). However, Applicants respectfully submit that the Examiner's allegation that Williams uses "parametric interpolation" to generate a true pixel color is **not** the same as the Applicants' claimed invention of "approximating a true pixel color by performing ... texturing operations, said texturing operations being determined by a geometric shape of a projection of a pixel on the texture." More specifically, "parametric interpolation" as disclosed by Williams may or may not generate a true pixel color, but Williams does not suggest, teach, or disclose a "parametric interpolation" that uses geometric shapes defined by a pixel's projection on a texture.

Furthermore, the Examiner stated, page 3, lines 2-3, that Williams discloses "averaging results of texturing operations (page 2; fig. 1)." First, as discussed above, Williams does not

disclose "texturing operations being determined by a geometric shape of a projection of a pixel on the texture," as claimed. Second, Williams discloses a structure of a color mipmap, where a smaller image represents an average of a larger image (FIG. 1 and FIG. 1 caption). However, the structure of a mipmap is not a texturing operation. That is, texturing operations may use mipmaps, but the structure of a mipmap is not a texturing operation. For example, in one embodiment of using mipmaps for texturing in conjunction with FIG. 2, the specification states, page 6, lines 15-18,

> Mipmapping is a reasonable candidate for a hardware implementation due to its regular access function. If the memory is designed to deliver all eight texels for a tri-linear interpolation in a single access, texturing can potentially keep up with fast rasterizer units.

Based on at least the above remarks, Applicants respectfully submit that claim 58 is not anticipated by Williams and request that claim 58 be allowed. Furthermore, since claims 59-64 depend directly or indirectly from claim 58, Applicants submit that claims 59-64 are allowable for at least the same reasons as claim 58.

With regard to claim 65, the Examiner stated on page 4, lines 1-5, that "Williams teaches modifying a specularly reflected light intensity on a surface" by "combining the specularly reflected light intensity with a specular reflectance coefficient ... retrieved from a specular reflectance coefficient map associated with the surface." However, Applicants respectfully submit that Williams (1) does not **modify** a specularly reflected light intensity on a surface, and (2) does not **combine** a specularly reflected light intensity with a specular reflectance coefficient. Williams discloses that a surface which point samples an illumination function generates illumination aliasing, resulting in shading values associated with a surface in motion to flash annoyingly, (page 7, first paragraph). In addition, Williams discloses that point-sampling a

specular surface reflection function (i.e., a highlight illumination function) may result in "highlight" aliasing, (page 7, paragraph 4; and FIG. 13). In order to overcome the limitations associated with point-sampling an illumination function, Williams proposes to integrate the illumination function over a projected area represented by each sample point, (page 7, paragraph 5; and Fig. 14). Williams then generalizes this antialiasing approach to curved surfaces by modifying the sample interval (i.e., projected area) over which the illumination function is to be integrated to a local curvature of the surface at a sample, (page 7, paragraph 6, lines 1-8; and Fig. 15). Thus, Williams does not modify a "specularly reflected light intensity on a surface" as claimed, but in contrast modifies a sample integration interval over which the illumination function is to be integrated.

Williams discloses a pyramidal parametric reflectance map to permit antialiasing of highlights, (page 8, paragraph 3; and Fig. 17). Although Williams may or may not generate specular reflectance coefficients from the pyramidal parametric reflectance map, Williams does not teach, suggest, or disclose a system or method for "combining the specularly reflected light intensity with a specular reflectance coefficient," as claimed, nor does the Examiner provide any evidence to the contrary.

Based on at least the above remarks, Applicants respectfully submit that claim 65 is not anticipated by Williams and request that claim 65 be allowed. Furthermore, since claim 66 depends from claim 65, and since claim 72 is similar in scope to claim 65, Applicants submit that claims 66 and 72 are allowable for at least the same reasons as claim 65.

With regard to claim 71, the Examiner stated, page 4, second paragraph, that "the limitations of claim 71 are identical to claim 58 above except for an electronically-readable medium storing a program for permitting a computer to perform." Thus, based on at least the

above remarks in conjunction with claim 58, the Applicants respectfully request that the rejection to claim 71 be withdrawn and claim 71 be allowed.

On page 4, paragraph 3, the Examiner stated that Williams teaches claim 73. However, Applicants respectfully submit that Williams does not disclose "interpolating **detail color** based on the generated detail map" and "**combining** detail color with texture color to generate a **pixel color**" (emphasis added), as claimed. In contrast, Williams discloses a pyramidal parametric data structure the components of which are **not** detail color, but instead are spatial coordinates of the vertices of a rectangular mesh, (page 9 third paragraph). Williams uses the pyramidal parametric data structure of spatial coordinates to limit the level of detail with which to represent a surface, (page 9, paragraph 1; and Figs. 20-23). However, Williams does not use the pyramidal parametric data structure of spatial coordinates for "interpolating detail color," as claimed. In addition, Williams does not suggest, teach, or disclose combining detail color obtained from a detail map (i.e., obtained from Williams' pyramidal parametric data structure of spatial coordinates) with a texture color to generate a pixel color.

Furthermore, the Examiner stated, page 4, paragraph 3, that "Williams teaches ... assigning a pointer (page 2; index corresponding to pointer) ... to at least one of the texture elements of the texture map to generate a pointer map ...." Applicants respectfully submit that Williams does not teach "assigning a pointer ... to at least one of the texture elements of the texture map to generate a pointer map," as claimed. On page 2, in the caption associated with the lower figure, Williams discloses a variable D used to index between the different levels of the pyramid. The index D is used to locate a position between the levels of the pyramid, however the index D is **not** a pointer assigned to at least one of the texture elements. For example, Williams does not disclose assigning a pointer to at least one of the texture elements as

illustrated in Figure 1. In contrast, FIG. 11(a) of the specification illustrates pointers assigned to texture elements (i.e., texels) in one embodiment of the Applicants' invention.

Based on at least the above remarks, Applicants respectfully submit that claim 73 is not anticipated by Williams and request that claim 73 be allowed.

Rejection under 35 U.S.C. § 103

On page 5, paragraph 2, the Examiner rejected claims 67-70 under 35 U.S.C. § 103(a) as being unpatentable over Williams in view of Cosman (U.S. Patent No. 5,651,104). Applicants respectfully traverse.

With regard to claim 67, the Examiner stated, page 5, paragraph 3, that Williams disclosure of "levels of detail in surface representation and dividing the surface up into regions of relatively low curvature" is the same as Applicants' invention of "determining a set of N footprint texel locations and at least one footprint level of detail from the input signal, which input signal includes information about a location and shape of a projection of a pixel on the texture," as claimed. As discussed above in conjunction with claim 58, Williams does not disclose projecting a pixel on a texture. Consequently, Applicants respectfully submit that Williams does not disclose an input signal that includes information about a location and shape of a projection of a pixel on a texture, nor has the Examiner provided any evidence to the contrary. That is, Applicants fail to see how dividing a surface up into regions of relatively low curvature (Williams, page 9 and FIGS. 20-23) is the same as projecting a pixel on a **texture**. A texture is not a surface or an object, but as stated in the specification, page 2, lines 1-2,

> ... we use the term "texture as a synonym for any image or structure to be mapped onto an object ....

Textures as disclosed by the specification include mipmaps comprised of a plurality of texels, where the texel values include, but are not limited to, colors and reflectances. In the Applicants' invention, pixels are projected on the textures, in contrast to dividing a surface up into regions of low curvature, as disclosed by <u>Williams</u>. Further, <u>Cosman</u> does not cure the defects of <u>Williams</u>.

Based on at least the above remarks, Applicants respectfully submit that claim 67 is not obvious over <u>Williams</u> in view of <u>Cosman</u>, and request that claim 67 be allowed. Furthermore, since claims 68-70 depend directly or indirectly from claim 67, Applicants submit that claims 68-70 are allowable for at least the same reasons as claim 67.

Based on the foregoing remarks, Applicants believe that the rejections and objections in the Office Action of February 14, 2003 are fully overcome, and that the Application is in condition for allowance. If the Examiner has questions regarding the case, he is invited to contact Applicants' undersigned representative at the number given below.

Respectfully submitted,

Andreas Schilling et al.

Date: 4/16/03

By: Susan Yee

Susan Yee, Reg. No. 41,388
Carr & Ferrell LLP
2225 East Bayshore Road, Suite 200
Palo Alto, CA 94303
Phone: (650) 812-3423
Fax: (650) 812-3444

APPENDIX

In the specification:

Paragraph beginning on page 7, line 4:

The rasterizer 302 computes texture coordinates (u,v) for each pixel. Texturing unit 306 receives the texture coordinates (u,v) of a pixel from rasterizer 302 over line 308 and retrieves a plurality of texels from the texture memory and interpolates the pixel's texture color (RGB) from the texels values. The term "line" as used herein is intended to refer generally to functional coupling of signals between logical blocks. As such the term "line" may refer to a single physical signal, or to a plurality of signals such as a bus. Rasterizer 302 receives the pixel's texture color from texturing unit 306 over line 310. The final pixel color (RGB) together with the z-value is stored in frame buffer 312 at address (x,y). Data stored in frame buffer 3[08]<u>12</u> may be subsequently used by rasterizer 302 for further operations in addition to being converted to analog form for display on a visula display unit (not shown) such as a Cathode Ray Tube (CRT) or Liquid Crystal Display (LCD). A description of two preferred embodiments of texturing unit 306 is provided below in Sections 4 and 5.

Paragraph beginning on page 13, line 9:

The system 802 includes memory banks 810 (designated individually as Bank 0, 1, 2, . . . 7), address and control unit 812, Color LookUp Table (CLUT) 814, mipmap generation unit 816, tri-linear interpolator 818 and output/combination stage 820. The capacity of memory [arrays 704] <u>banks 810</u> sum up to 11,239,424 bits, and thus, in a preferred embodiment, a 16Mbit DRAM technology is used. The memory system

consists of four large arrays (Banks 0, 1, 2 and 3) of 274x8192 bits, holding the even levels of the mipmap, and four small arrays of 69x8192 bits (Banks 4, 5, 6 and 7) for the odd levels of the mipmap. Control unit 812 is advantageously pipelined and includes a plurality of Description Register Files (DRFs), explained below in Section 14. The control unit 812 generates all addresses and controls internal operation and the flow of data to and from a rasterizer such as shown in Figure 3. The tri-linear interpolator 818 is designed for a 6-bit fraction of the texture coordinates.

Paragraph beginning on page 14, line 12:

Address decoder 830 and CLUT 814 allow a parallel look-up of eight color values at a time. The output of CLUT 814 is coupled via lines 832 to the input of tri-linear interpolator 818. Lines 828, as mentioned above, are also coupled to the input of tri-linear interpolator 818, allowing a bypass of CLUT 814 in instances where the contents of memory banks 810 hold true color data as opposed to addresses for CLUT 814. Tri-linear interpolator <u>818</u> performs a tri-linear interpolation and provides a color value to output and combination stage 820 which implements the functions described below in sections 6-10. The output of the stage [6]<u>8</u>20 is coupled to the rasterizer 302.